## CLAIMS

Please amend claims 1, 3, 5, 7, and 8 as follows:

1. (Currently amended) A processing system for performing addition and subtraction within limits upon a shared value comprising:

means for performing a first uninterruptible operation upon the shared value stored in an affected reservation location, the first uninterruptible operation using an operand;

means for comparing a resulting value of the first uninterruptible operation stored in the affected reservation location to limit values stored in limit locations;

means for performing a second uninterruptible operation to restore the affected reservation location if the resulting value of the first uninterruptible operation is not within the limit values in the limit locations;

means for reporting a failure if the resulting value of the first uninterruptible operation is not within the limit values in the limit locations;

means for performing a third uninterruptible operation to update an actual value location if the resulting value of the first uninterruptible operation is within the limit values in the limit locations;

means for performing a fourth uninterruptible operation to update an unaffected reservation location [register] if the resulting value of the first uninterruptible operation is within the limit values in the limit locations; and

means for reporting a success if the resulting value of the first uninterruptible operation is within the limit values in the limit locations.

2. (Previously presented) The processing system of claim 1 wherein the first, second, third, and fourth uninterruptible operations are LOCK XADD operations.

3. (Currently amended) A processing system for performing addition and subtraction within limits upon a shared value comprising:

means for receiving an operand;

means for performing a first uninterruptible operation upon the shared value stored in an affected reservation location, the first uninterruptible operation using the operand;

means for comparing a resulting value of the first uninterruptible operation stored in the affected reservation location to limit values stored in limit locations;

means for performing a second uninterruptible operation to restore the affected reservation location if the resulting value of the first uninterruptible operation is not within the limit values in the limit locations;

means for and reporting a failure if the resulting value of the first uninterruptible operation is not within the limit values in the limit locations;

means for performing a third uninterruptible operation to update an actual value location if the resulting value of the first uninterruptible operation is within the limit values in the limit locations;

means for performing a fourth uninterruptible operation to update an unaffected reservation location [register] if the resulting value of the first uninterruptible operation is within the limit values in the limit locations; and

means for reporting a success if the resulting value of the first uninterruptible operation is within the limit values in the limit locations.

4. (Previously presented) The processing system of claim 3 wherein the first, second, third, and fourth uninterruptible operations are LOCK XADD operations.

5. (Currently amended) A method for performing addition and subtraction within limits upon a shared value comprising the steps of:

first, performing a first uninterruptible operation upon the shared value stored in an affected reservation location, the first uninterruptible operation using an operand;

second, comparing a resulting value of the first uninterruptible operation stored in the affected reservation location to limit values stored in limit locations;

third, performing a second uninterruptible operation to restore the affected reservation location;

fourth, reporting a failure if the resulting value is not within the limit values in the limit locations;

fifth, performing a third uninterruptible operation to update an actual value location if the resulting value is within the limit values in the limit locations;

sixth, performing a fourth uninterruptible operation to update an unaffected reservation location [register] if the resulting value is within the limit values in the limit locations; and

seventh, reporting a success if the resulting value is within the limit values in the limit locations.

6. (Previously presented) The method of claim 5 wherein the first, second, third, and fourth uninterruptible operations are LOCK XADD operations.

7. (Currently amended) A computer readable medium containing computer readable code comprising:

a code segment for performing a first uninterruptible operation upon the shared value stored in an affected reservation location, the first uninterruptible operation using an operand;

a code segment for comparing a resulting value of the first uninterruptible operation stored in the affected reservation location to limit values stored in limit locations;

a code segment for performing a second uninterruptible operation to restore the affected reservation location;

a code segment for reporting a failure if the resulting value is not within the limit values in the limit locations;

a code segment for performing a third uninterruptible operation to update an actual value location if the resulting value is within the limit values in the limit locations;

a code segment for performing a fourth uninterruptible operation to update an unaffected reservation location [register] if the resulting value is within the limit values in the limit locations; and

a code segment for reporting a success if the resulting value is within the limit values in the limit locations.

8. (Currently amended) A processing system for performing addition and subtraction within limits upon a shared value comprising:

a processor, the processor

performing a first uninterruptible operation upon the shared value stored in an affected reservation location, the first uninterruptible operation using an operand;

comparing a resulting value of the first uninterruptible operation stored in the affected reservation location to limit values stored in limit locations;

performing a second uninterruptible operation to restore the affected reservation location if the resulting value of the first uninterruptible operation is not within the limit values in the limit locations;

reporting a failure if the resulting value of the first uninterruptible operation is not within the limit values in the limit locations;

performing a third uninterruptible operation to update an actual value location if the resulting value of the first uninterruptible operation is within the limit values in the limit locations;

performing a fourth uninterruptible operation to update an unaffected reservation location [register] if the resulting value of the first uninterruptible operation is within the limit values in the limit locations; and

reporting a success if the resulting value of the first uninterruptible operation is within the limit values in the limit locations.